Amendment to the Claims

Kindly amend claims 1, 3, 12, 17, 22, 24, 33, 38, 43, 45, 47, 49, 51, 60 & 65 as set forth below. In compliance with the Revised Amendment Format published in the Official Gazette on February 25, 2003, a complete listing of claims is provided herein. The changes in the amended claims are shown by strikethrough (for deleted matter) and underlining (for added matter).

---

1. (Currently Amended) A method of managing the prefetching of data of files, said method comprising:

A1

detecting a pattern of requests for data of multiple files, wherein the pattern is based on one or more user-defined attributes of the multiple files; and

prefetching data of a plurality of files, in response to said detecting indicating said pattern.

2. (Original) The method of claim 1, wherein said data comprises meta data.

3. (Currently Amended) The method of claim 1, wherein said multiple files and said plurality of files are ~~associated with~~ within a single directory.

4. (Original) The method of claim 1, wherein said detecting indicates said pattern when said detecting determines that a predefined number of requests for data could not be satisfied by reading a cache.

5. (Original) The method of claim 1, wherein said detecting comprises determining whether a cache miss threshold has been exceeded, wherein said detecting indicates said pattern when said cache miss threshold has been exceeded.

6. (Original) The method of claim 5, wherein said determining comprises comparing a counter of cache misses that occurred within a preselected time interval to said cache miss threshold to determine whether said cache miss threshold has been exceeded.

7. (Original) The method of claim 6, wherein said counter and said cache miss threshold are associated with a directory block of a directory of files, said directory of files comprising said multiple files and said plurality of files, and said directory of files comprising one or more directory blocks.

8. (Original) The method of claim 1, wherein said prefetching comprises prefetching data of at least some files of said plurality of files in parallel.

A1

9. (Original) The method of claim 1, wherein said prefetching comprises prefetching data at an average rate that substantially matches a speed of requests for data.

10. (Original) The method of claim 1, wherein said prefetching comprises:

obtaining data associated with a number of files of said plurality of files;

determining whether a cache hit threshold has been reached; and

obtaining data associated with one or more additional files of said plurality of files, in response to reaching said cache hit threshold.

11. (Original) The method of claim 10, wherein said obtaining data associated with said number of files comprises issuing a plurality of I/O requests to read data of at least a portion of said number of files in parallel.

12. (Currently Amended) A method of managing the prefetching of data, said method comprising:

controlling, subsequent to determining that prefetching of data is to occur, a rate at which data of a plurality of files is prefetched by pacing ~~the prefetching of data of a plurality of files by pacing at least the initiating of~~ the prefetching based upon requests for data; and

prefetching said ~~meta~~ data of said plurality of files, in response to said controlling.

13. (Original) The method of claim 12, wherein said controlling comprises determining whether a cache hit threshold has been reached, wherein said prefetching is performed in response to reaching said cache hit threshold.

A1

14. (Original) The method of claim 12, wherein said prefetching comprises prefetching data of at least some files of said plurality of files in parallel.

15. (Original) The method of claim 12, further comprising detecting a pattern of requests for data of multiple files, wherein said pattern indicates prefetching is to occur.

16. (Original) The method of claim 12, wherein said data comprises meta data.

17. (Currently Amended) A method of managing the prefetching of inodes associated with files of a directory, said directory comprising one or more directory blocks and each directory block having associated therewith zero or more files, said method comprising:

detecting a pattern of requests for multiple inodes associated with multiple files of a directory block of said one or more directory blocks, wherein the pattern is based on directory entries of the multiple files being within said directory block; and

prefetching a plurality of inodes associated with said directory block, in response to detecting said pattern.

18. (Original) The method of claim 17, wherein said directory block has associated therewith a counter and a cache miss threshold, said counter representing a number of inodes associated with said directory block that were requested within a preselected amount of time and were not found in a cache, and wherein said detecting comprises comparing said counter to said cache miss threshold to determine whether said pattern exists.

19. (Original) The method of claim 17, wherein said prefetching comprises prefetching at least a portion of said plurality of inodes in parallel.

20. (Original) The method of claim 17, further comprising initiating the prefetching of one or more inodes associated with another directory block of said directory, wherein said initiating is in response to requests for inodes of said directory.

A1

21. (Original) The method of claim 20, wherein said initiating comprises determining whether a cache hit threshold has been reached, wherein said prefetching of one or more inodes associated with said another directory block is initiated when said cache hit threshold is reached.

22. (Currently Amended) A system of managing the prefetching of data of files, said system comprising:

means for detecting a pattern of requests for data of multiple files, wherein the pattern is based on one or more user-defined attributes of the multiple files; and

means for prefetching data of a plurality of files, in response to said means for detecting indicating said pattern.

23. (Original) The system of claim 22, wherein said data comprises meta data.

24. (Currently Amended) The system of claim 22, wherein said multiple files and said plurality of files are within ~~associated with~~ a single directory.

25. (Original) The system of claim 22, wherein said means for detecting indicates said pattern when said means for detecting determines that a predefined number of requests for data could not be satisfied by reading a cache.

26. (Original) The system of claim 22, wherein said means for detecting comprises means for determining whether a cache miss threshold has been exceeded, wherein said means for detecting indicates said pattern when said cache miss threshold has been exceeded.

27. (Original) The system of claim 26, wherein said means for determining comprises means for comparing a counter of cache misses that occurred within a preselected time interval to said cache miss threshold to determine whether said cache miss threshold has been exceeded.

28. (Original) The system of claim 27, wherein said counter and said cache miss threshold are associated with a directory block of a directory of files, said directory of files comprising said multiple files and said plurality of files, and said directory of files comprising one or more directory blocks.

29. (Original) The system of claim 22, wherein said means for prefetching comprises means for prefetching data of at least some files of said plurality of files in parallel.

30. (Original) The system of claim 22, wherein said means for prefetching comprises means for prefetching data at an average rate that substantially matches a speed of requests for data.

31. (Original) The system of claim 22, wherein said means for prefetching comprises:

means for obtaining data associated with a number of files of said plurality of files;

means for determining whether a cache hit threshold has been reached; and

means for obtaining data associated with one or more additional files of said plurality of files, in response to reaching said cache hit threshold.

32. (Original) The system of claim 31, wherein said means for obtaining data associated with said number of files comprises means for issuing a plurality of I/O requests to read data of at least a portion of said number of files in parallel.

33. (Currently Amended) A system of managing the prefetching of data, said system comprising:

A1

means for controlling, subsequent to determining that prefetching of data is to occur, a rate at which data of a plurality of files is prefetched by pacing ~~the prefetching of data of a plurality of files by pacing at least the initiating of~~ the prefetching based upon requests for data; and

means for prefetching said data of said plurality of files, in response to the controlling.

34. (Original) The system of claim 33, wherein said means for controlling comprises means for determining whether a cache hit threshold has been reached, wherein prefetching is performed in response to reaching said cache hit threshold.

35. (Original) The system of claim 33, wherein said means for prefetching comprises means for prefetching data of at least some files of said plurality of files in parallel.

36. (Original) The system of claim 33, further comprising means for detecting a pattern of requests for data of multiple files, wherein said pattern indicates prefetching is to occur.

37. (Original) The system of claim 33, wherein said data comprises meta data.

38. (Currently Amended) A system of managing the prefetching of inodes associated with files of a directory, said directory comprising one or more directory blocks and each directory block having associated therewith zero or more files, said system comprising:

means for detecting a pattern of requests for multiple inodes associated with multiple files of a directory block of said one or more directory blocks, wherein the pattern is based on directory entries of the multiple files being within said directory block; and

A1

means for prefetching a plurality of inodes associated with said directory block, in response to detecting said pattern.

39. (Original) The system of claim 38, wherein said directory block has associated therewith a counter and a cache miss threshold, said counter representing a number of inodes associated with said directory block that were requested within a preselected amount of time and were not found in a cache, and wherein said means for detecting comprises means for comparing said counter to said cache miss threshold to determine whether said pattern exists.

40. (Original) The system of claim 38, wherein said means for prefetching comprises means for prefetching at least a portion of said plurality of inodes in parallel.

41. (Original) The system of claim 38, further comprising means for initiating the prefetching of one or more inodes associated with another directory block of said directory, wherein the initiating is in response to requests for inodes of said directory.

42. (Original) The system of claim 41, wherein said means for initiating comprises means for determining whether a cache hit threshold has been reached, wherein the prefetching of one or more inodes associated with said another directory block is initiated when said cache hit threshold is reached.

43. (Currently Amended) A system of managing the prefetching of data of files, said system comprising:

a first node adapted to detect a pattern of requests for data of multiple files wherein the pattern is based on one or more user-defined attributes of the multiple files; and

at least one second node adapted to prefetch data of a plurality of files, in response to the detecting indicating said pattern.

A1

44. (Original) The system of claim 43, wherein said at least one second node includes said first node.

45. (Currently Amended) A system of managing the prefetching of data, said system comprising:

a first node adapted to control, subsequent to determining that prefetching of data is to occur, a rate at which data of a plurality of files is prefetched by pacing ~~the prefetching of data of a plurality of files by pacing at least the initiating of~~ the prefetching based upon requests for data; and

at least one second node adapted to prefetch said data of said plurality of files, in response to the controlling.

46. (Original) The system of claim 45, wherein said at least one second node includes said first node.

47. (Currently Amended) A system of managing the prefetching of inodes associated with files of a directory, said directory comprising one or more directory blocks and each directory block having associated therewith zero or more files, said system comprising:

a first node adapted to detect a pattern of requests for multiple inodes associated with multiple files of a directory block of said one or more directory blocks, wherein the pattern is based on directory entries of the multiple files being within said directory block; and

at least one second node adapted to prefetch a plurality of inodes associated with said directory block, in response to detecting said pattern.

A1

48. (Original) The system of claim 47, wherein said at least one second node includes said first node.

49. (Currently Amended) At least one program storage device readable by a machine, tangibly embodying at least one program of instructions executable by the machine to perform a method of managing the prefetching of data of files, said method comprising:

detecting a pattern of requests for data of multiple files, wherein the pattern is based on one or more user-defined attributes of the multiple files; and

prefetching data of a plurality of files, in response to said detecting indicating said pattern.

50. (Original) The at least one program storage device of claim 49, wherein said data comprises meta data.

51. (Currently Amended) The at least one program storage device of claim 49, wherein said multiple files and said plurality of files are within ~~associated with~~ a single directory.

52. (Original) The at least one program storage device of claim 49, wherein said detecting indicates said pattern when said detecting determines that a predefined number of requests for data could not be satisfied by reading a cache.

53. The at least one program storage device of claim 49, wherein said detecting comprises determining whether a cache miss threshold has been exceeded, wherein said detecting indicates said pattern when said cache miss threshold has been exceeded.

54. (Original) The at least one program storage device of claim 53, wherein said determining comprises comparing a counter of cache misses that occurred within a preselected time interval to said cache miss threshold to determine whether said cache miss threshold has been exceeded.

A1

55. (Original) The at least one program storage device of claim 54, wherein said counter and said cache miss threshold are associated with a directory block of a directory of files, said directory of files comprising said multiple files and said plurality of files, and said directory of files comprising one or more directory blocks.

56. (Original) The at least one program storage device of claim 49, wherein said prefetching comprises prefetching data of at least some files of said plurality of files in parallel.

57. (Original) The at least one program storage device of claim 49, wherein said prefetching comprises prefetching data at an average rate that substantially matches a speed of requests for data.

58. (Original) The at least one program storage device of claim 49, wherein said prefetching comprises:

obtaining data associated with a number of files of said plurality of files;

determining whether a cache hit threshold has been reached; and

obtaining data associated with one or more additional files of said plurality of files, in response to reaching said cache hit threshold.

59. (Original) The at least one program storage device of claim 58, wherein said obtaining data associated with said number of files comprises issuing a plurality of I/O requests to read data of at least a portion of said number of files in parallel.

60. (Currently Amended) At least one program storage device readable by a machine, tangibly embodying at least one program of instructions executable by the machine to perform a method of managing the prefetching of data, said method comprising:

A1

controlling, subsequent to determining that prefetching of data is to occur, a rate at which data of a plurality of files is prefetched by pacing ~~the prefetching of data of a plurality of files by pacing at least the initiating of~~ the prefetching based upon requests for data; and

prefetching said data of said plurality of files, in response to said controlling.

61. (Original) The at least one program storage device of claim 60, wherein said controlling comprises determining whether a cache hit threshold has been reached, wherein said prefetching is performed in response to reaching said cache hit threshold.

62. (Original) The at least one program storage device of claim 60, wherein said prefetching comprises prefetching data of at least some files of said plurality of files in parallel.

63. (Original) The at least one program storage device of claim 60, wherein said method further comprises detecting a pattern of requests for data of multiple files, wherein said pattern indicates prefetching is to occur.

64. (Original) The at least one program storage device of claim 60, wherein said data comprises meta data.

65. (Currently Amended) At least one program storage device readable by a machine, tangibly embodying at least one program of instructions executable by the machine to perform a method of managing the prefetching of inodes associated with files of a directory, said directory having associated therewith one or more directory blocks and each directory block comprising zero or more files, said method comprising:

detecting a pattern of requests for multiple inodes associated with multiple files of a directory block of said one or more directory blocks, wherein the pattern is based on directory entries of the multiple files being within said directory block; and

A4

prefetching a plurality of inodes associated with said directory block, in response to detecting said pattern.

66. (Original) The at least one program storage device of claim 65, wherein said directory block has associated therewith a counter and a cache miss threshold, said counter representing a number of inodes associated with said directory block that were requested within a preselected amount of time and were not found in a cache, and wherein said detecting comprises comparing said counter to said cache miss threshold to determine whether said pattern exists.

67. (Original) The at least one program storage device of claim 65, wherein said prefetching comprises prefetching at least a portion of said plurality of inodes in parallel.

68. (Original) The at least one program storage device of claim 65, wherein said method further comprises initiating the prefetching of one or more inodes associated with another directory block of said directory, wherein said initiating is in response to requests for inodes of said directory.

69. (Original) The at least one program storage device of claim 68, wherein said initiating comprises determining whether a cache hit threshold has been reached, wherein said prefetching of one or more inodes associated with said another directory block is initiated when said cache hit threshold is reached.